Tenth DOE ACTS Collection Workshop

Leveraging the Development of Computational Science and Engineering Software Through Sustainable High Performance Tools

Tony Drummond
Computational Research Division
Lawrence Berkeley National Laboratory

Berkeley, California
August 2009

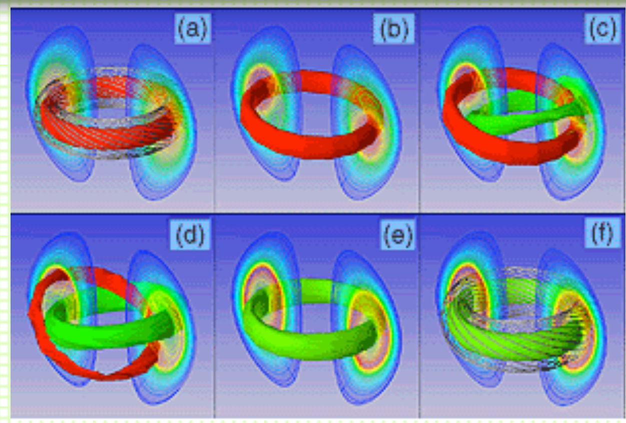
ACTS COLLECTION
BERKELEY LAB
Office of Science
U.S. DEPARTMENT OF ENERGY

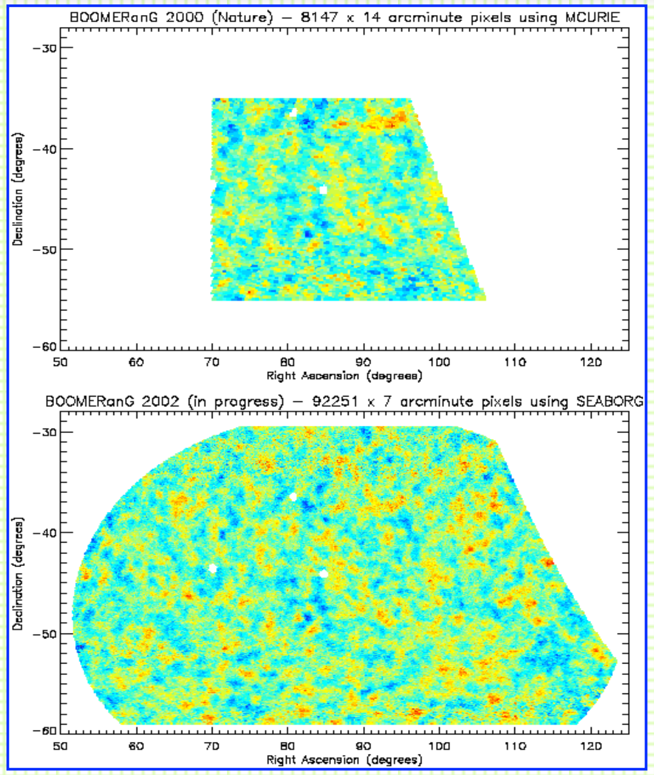
Outline

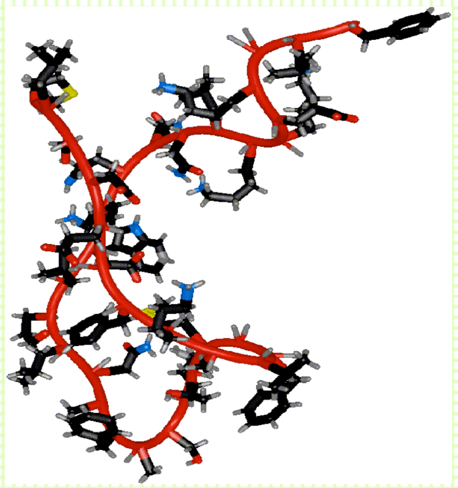
- Motivation
- Introduction to the DOE ACTS Collection
- Available functionality in the ACTS Collection
- Software sustainability
- This week at the ACTS Collection Workshop

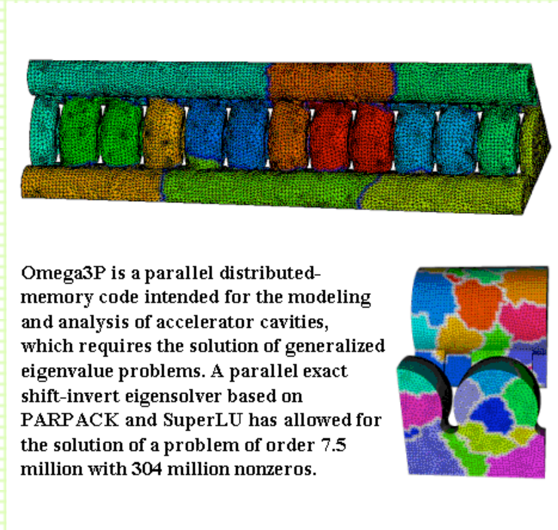
Motivation:
Computational Sciences and Engineering

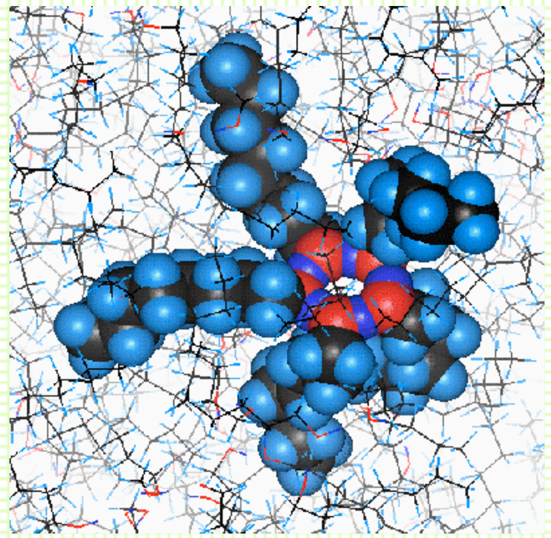
- Accelerator Science
- Astrophysics
- Biology
- Chemistry
- Earth Sciences
- Materials Science
- Nanoscience
- Plasma Science
-
-





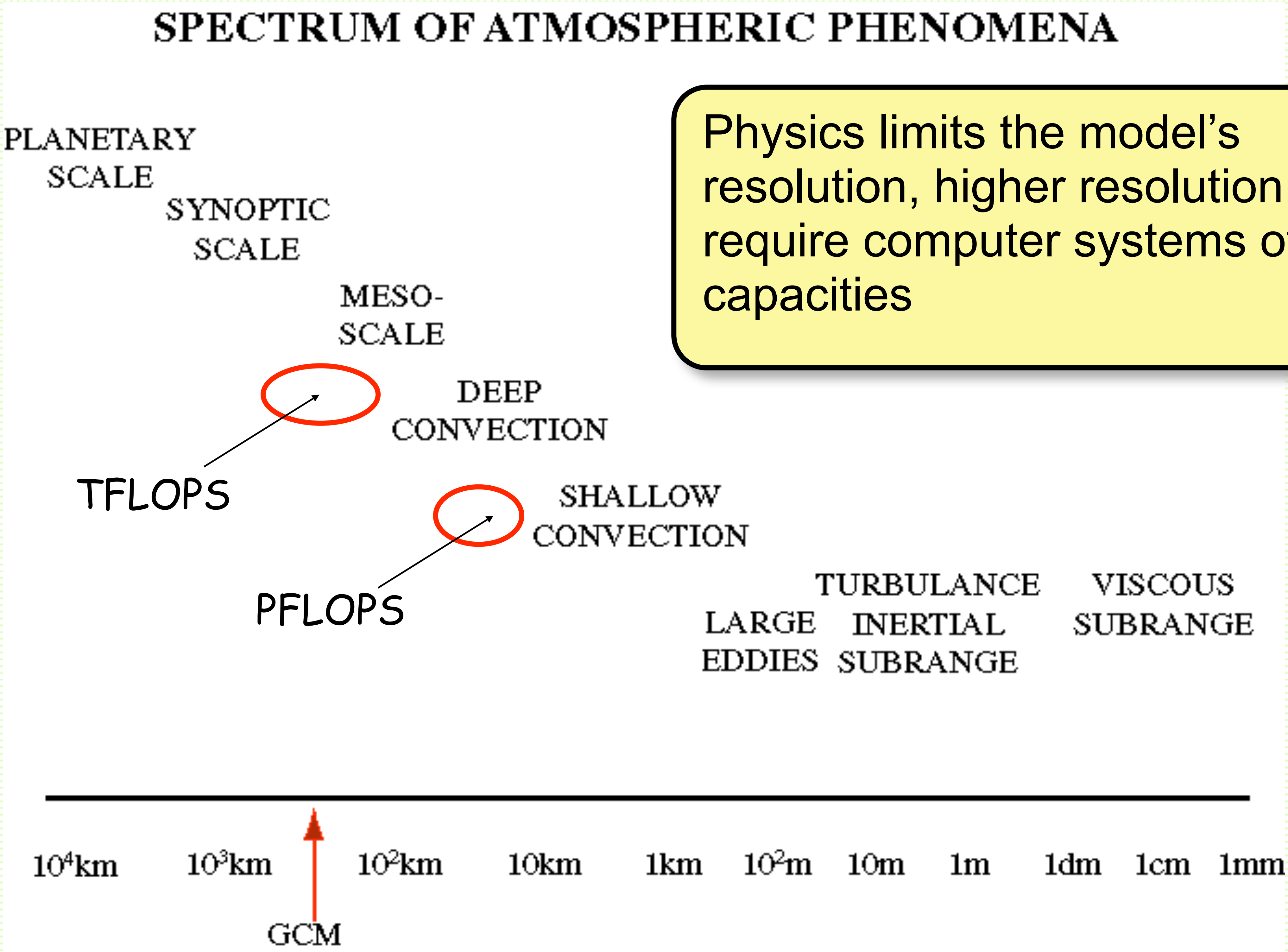






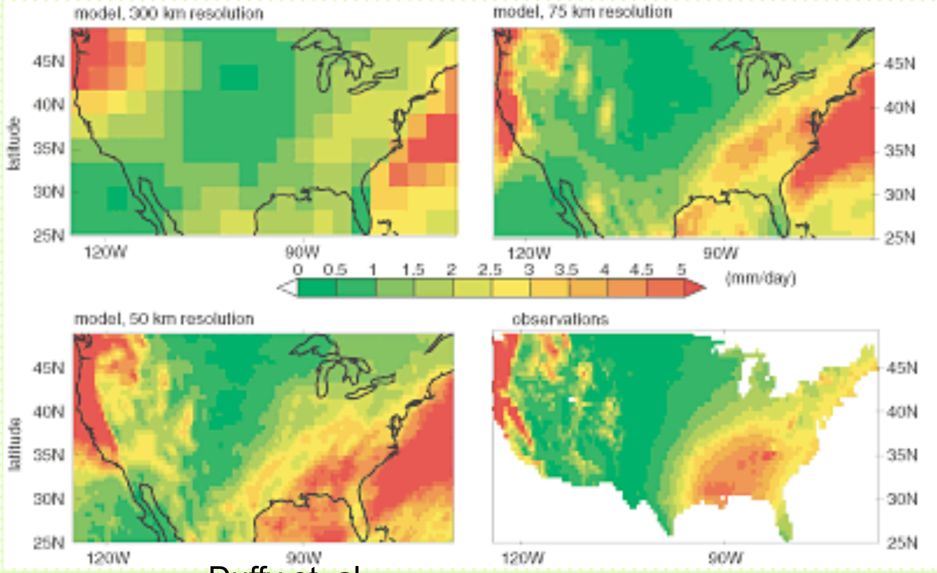
Commonalities:
- Major advancements in Science
- Increasing demands for computational power
- Rely on available computational systems, languages, and software tools

An Increasing Demand For FLOPS



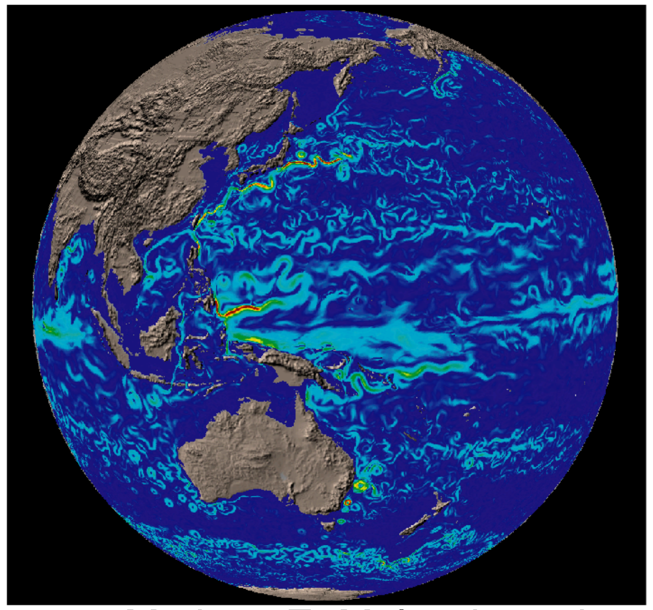
Physics limits the model's resolution, higher resolution will require computer systems of higher capacities

Examples of FLOP Demanding Applications



Duffy et. al.,
Lawrence Livermore National Laboratory

1/10 Degree Global POP Ocean Model Currents at 50m Depth
(blue = 0; red > 150 cm/s)



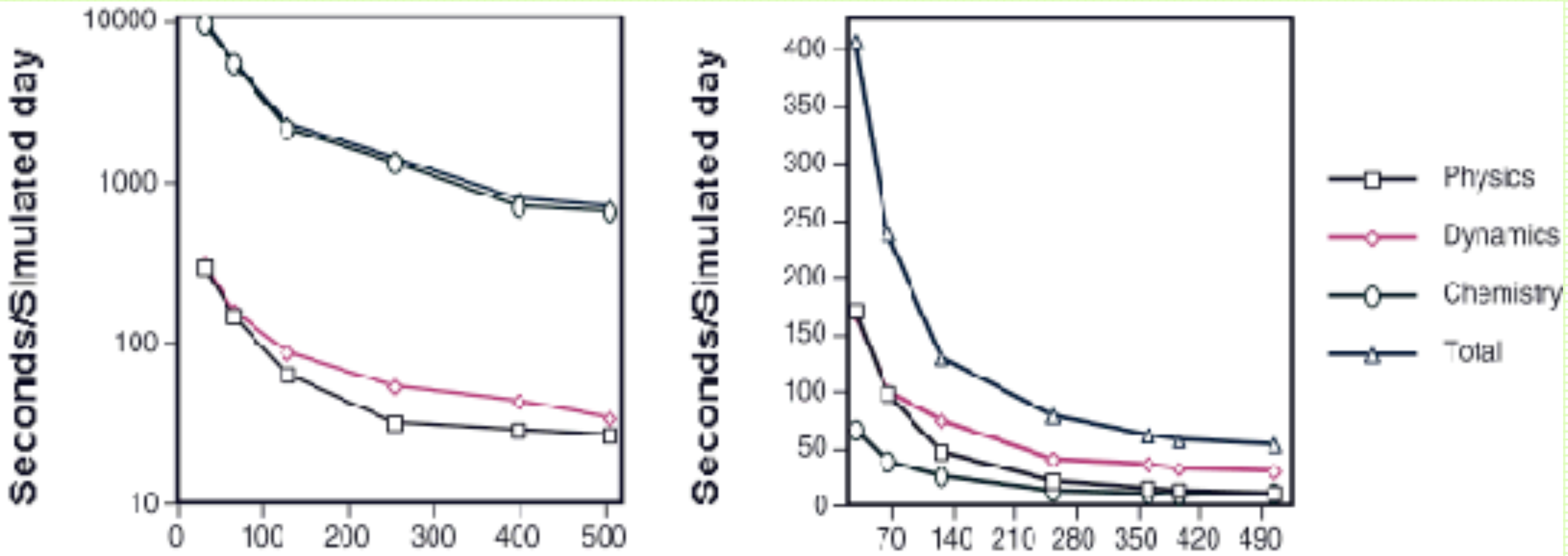
Mathew E. Maltruda and
Julie L. McClean

- Atmospheric general circulation model
  - Dynamics
  - Sub-grid scale parameterized physics processes
    - Turbulence, solar/infrared radiation transport, clouds.
- Oceanic general circulation model
  - Dynamics (mostly)
- Sea ice model
  - Viscous elastic plastic dynamics
  - Thermodynamics

- Land Model
  - Energy and moisture budgets
  - Biology
- Chemistry
  - Tracer advection, possibly stiff rate equations.
- Ocean Biology

Unpredictable Computational Demand

Climate Models:
- Different model resolutions have different computational demands
- Different model configurations may trigger different demands



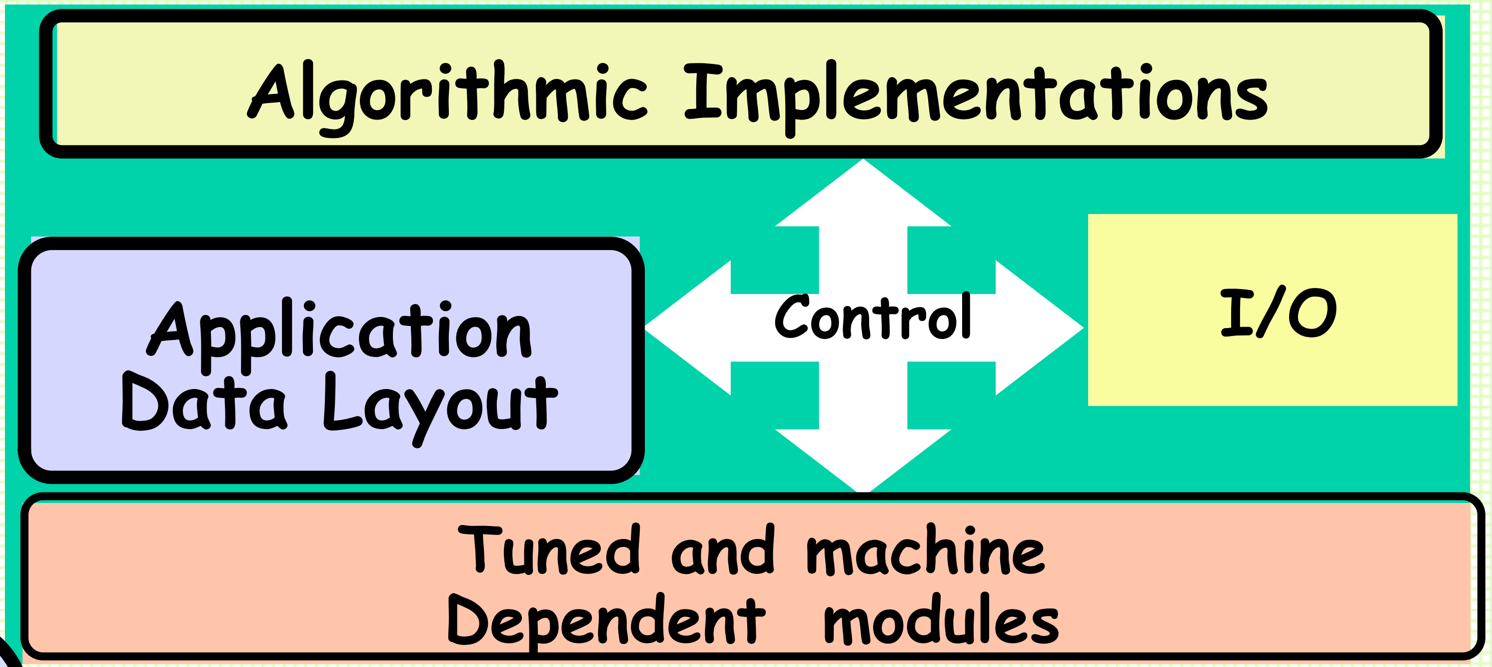
AGCM/ACM
2.5 long x 2 lat, 30 layers
25-chemical species

AGCM/ACM
2.5 long x 2 lat, 30 layers
2-chemical species

- Coupling of multi-domain and multi-resolution models are inherently load imbalanced

Programming and Software Abstractions

Changes in algorithms sometimes lead to several years advancement in computations. Needs Flexibility!



Its performance is influenced by system parameters and in steps in the algorithm. Critical points: portability and scalability.

New Architecture requires extensive tuning, may even require new programming paradigms. This is **Difficult to maintain and not "very" portable.**

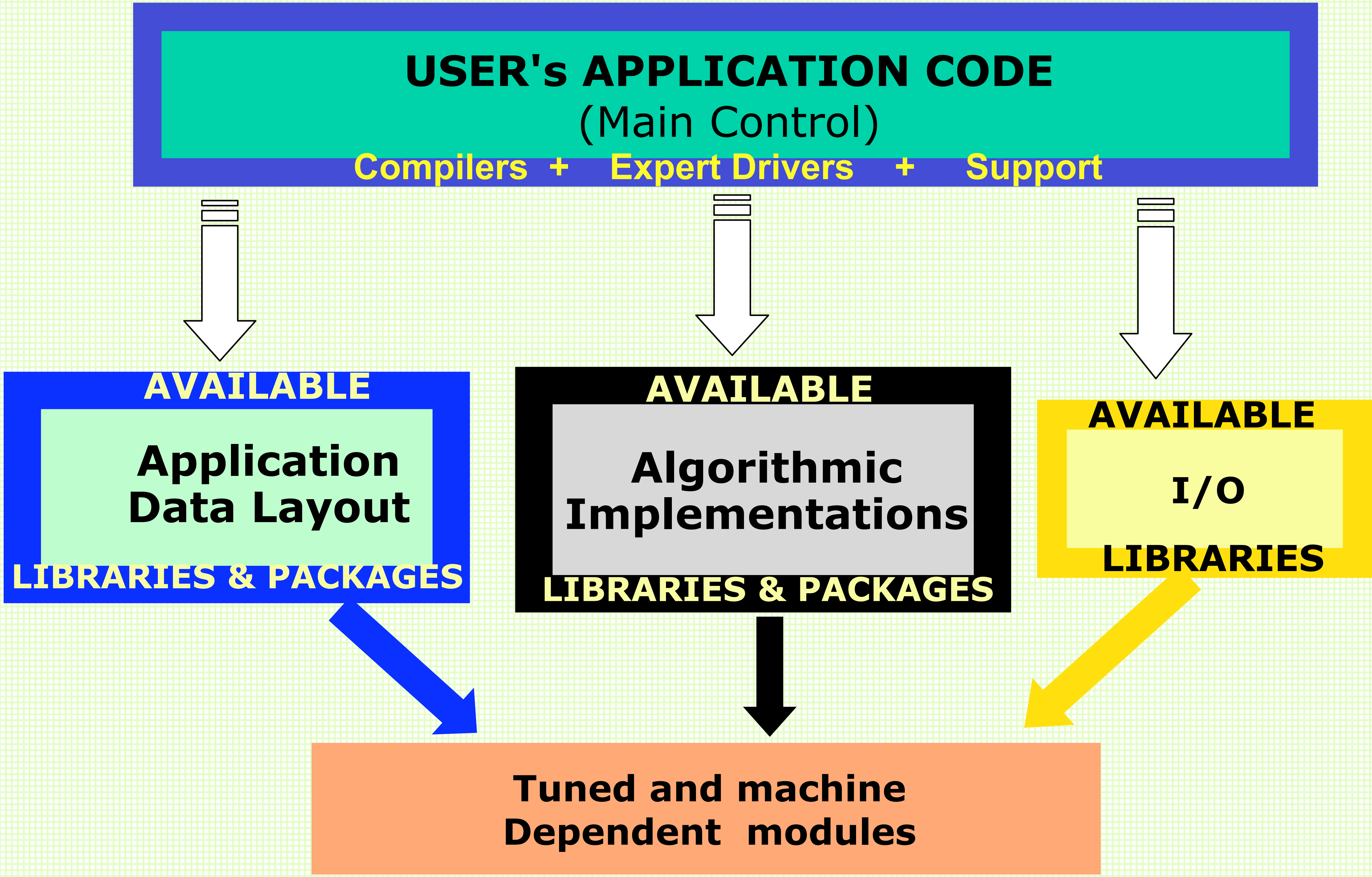
Programming and Software Abstractions

"We need to move away from a coding style suited for serial machines, where every macrostep of an algorithm needs to be thought about and explicitly coded, to a higher-level style, where the compiler and library tools take care of the details. And the remarkable thing is, if we adopt this higher-level approach right now, even on today's machines, we will see immediate benefits in our productivity."

W. H. Press and S. A. Teukolsky, 1997

Numerical Recipes: Does This Paradigm Have a future?

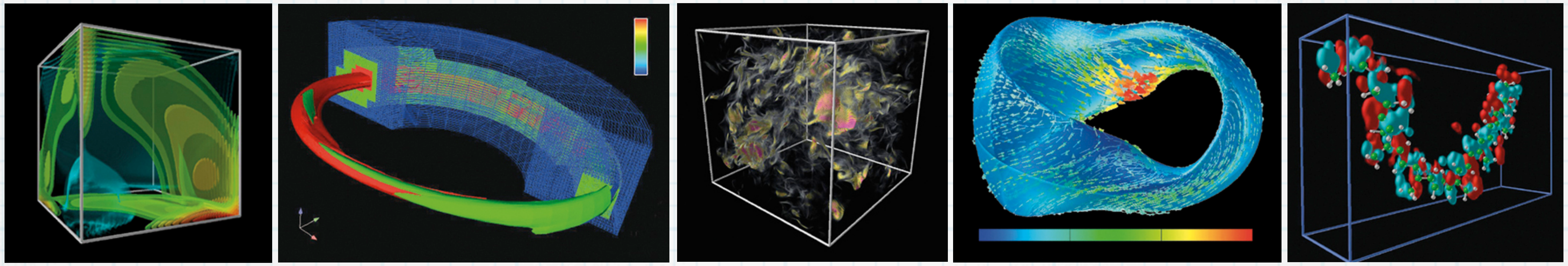
Programming and Software Abstractions



Outline

- Motivation
- Introduction to the DOE ACTS Collection

The U.S. DOE ACTS Collection



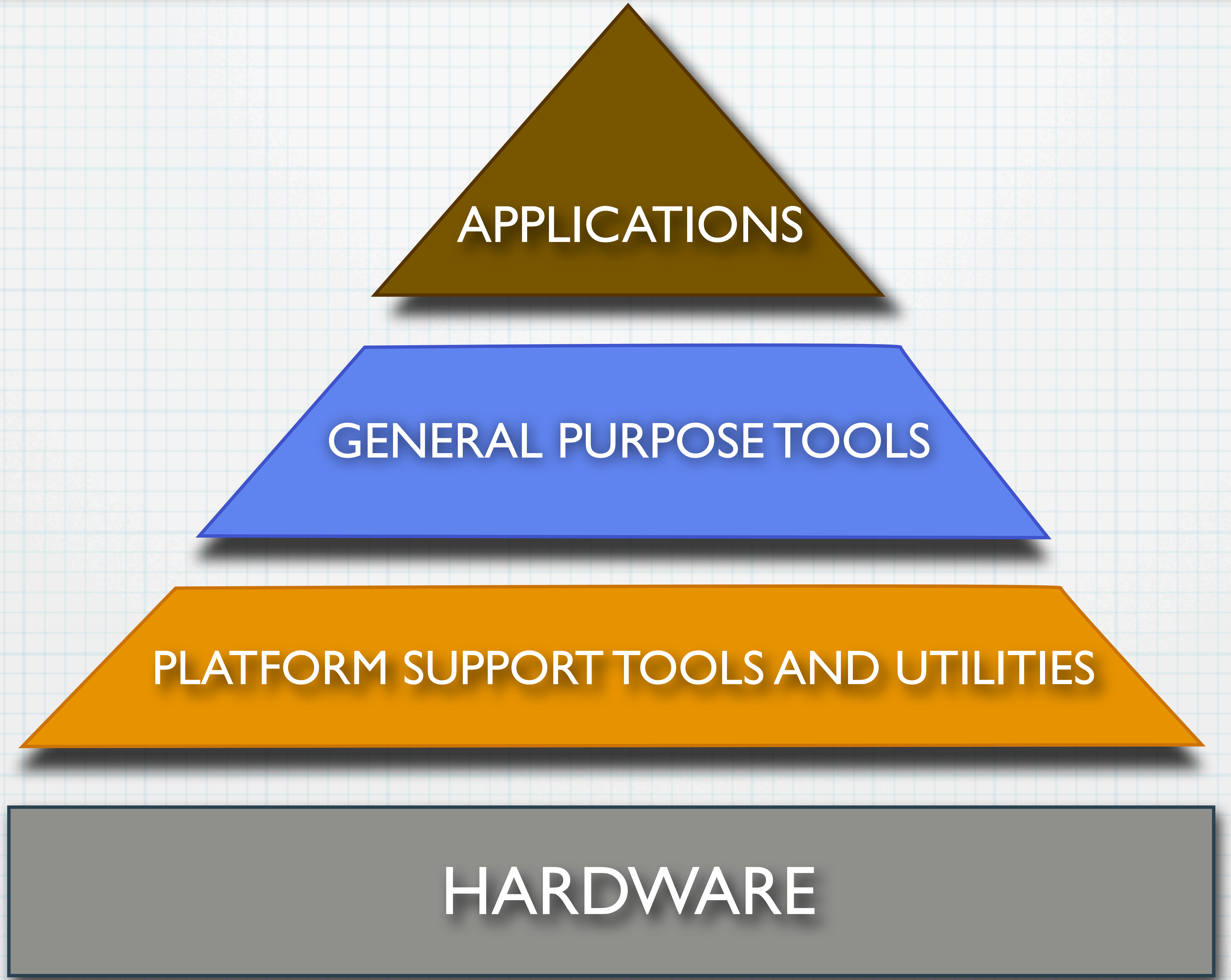
Goal: The Advanced CompuTational Software Collection (ACTS) makes reliable and efficient software tools more widely used, and more effective in solving the nation's engineering and scientific problems.

<u>References:</u>

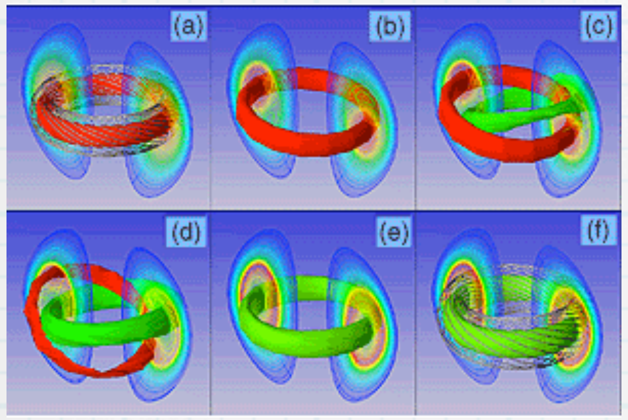
- L.A. Drummond, O. Marques: An Overview of the Advanced CompuTational Software (ACTS) Collection. ACM Transactions on Mathematical Software Vol. 31 pp. 282-301, 2005
- http://acts.nersc.gov

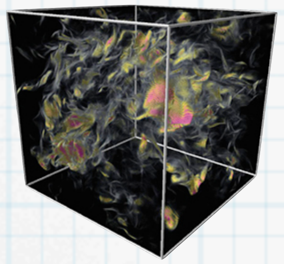
ACTS COLLECTION BERKELEY LAB Office of Science U.S. DEPARTMENT OF ENERGY

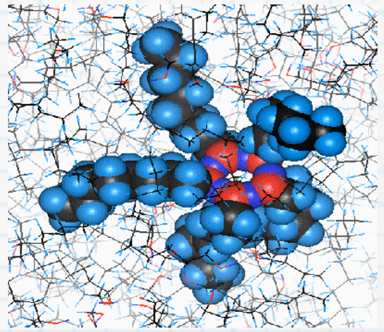
HPC Software Stack

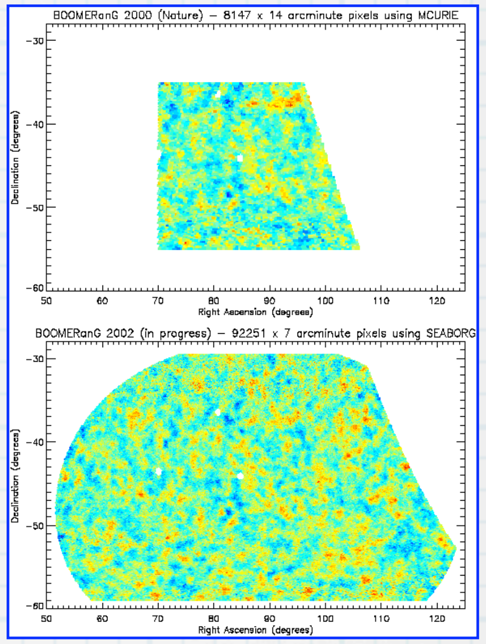


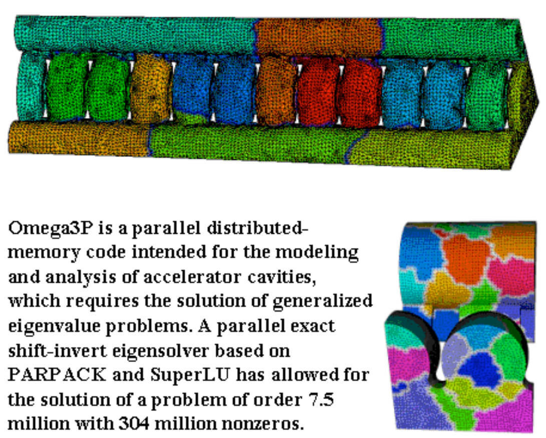
HPC Software Stack viewed from CSE





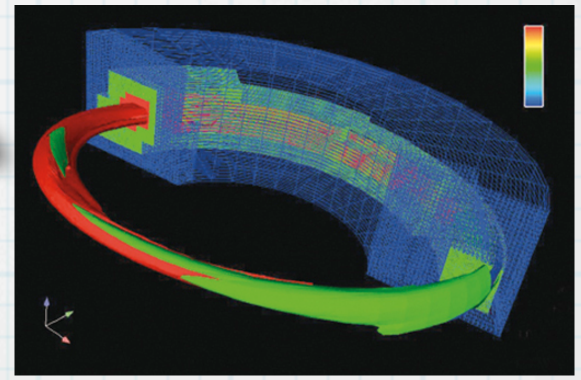






Omega3P is a parallel distributed-memory code intended for the modeling and analysis of accelerator cavities, which requires the solution of generalized eigenvalue problems. A parallel exact shift-invert eigensolver based on PARPACK and SuperLU has allowed for the solution of a problem of order 7.5 million with 304 million nonzeros.

APPLICATIONS

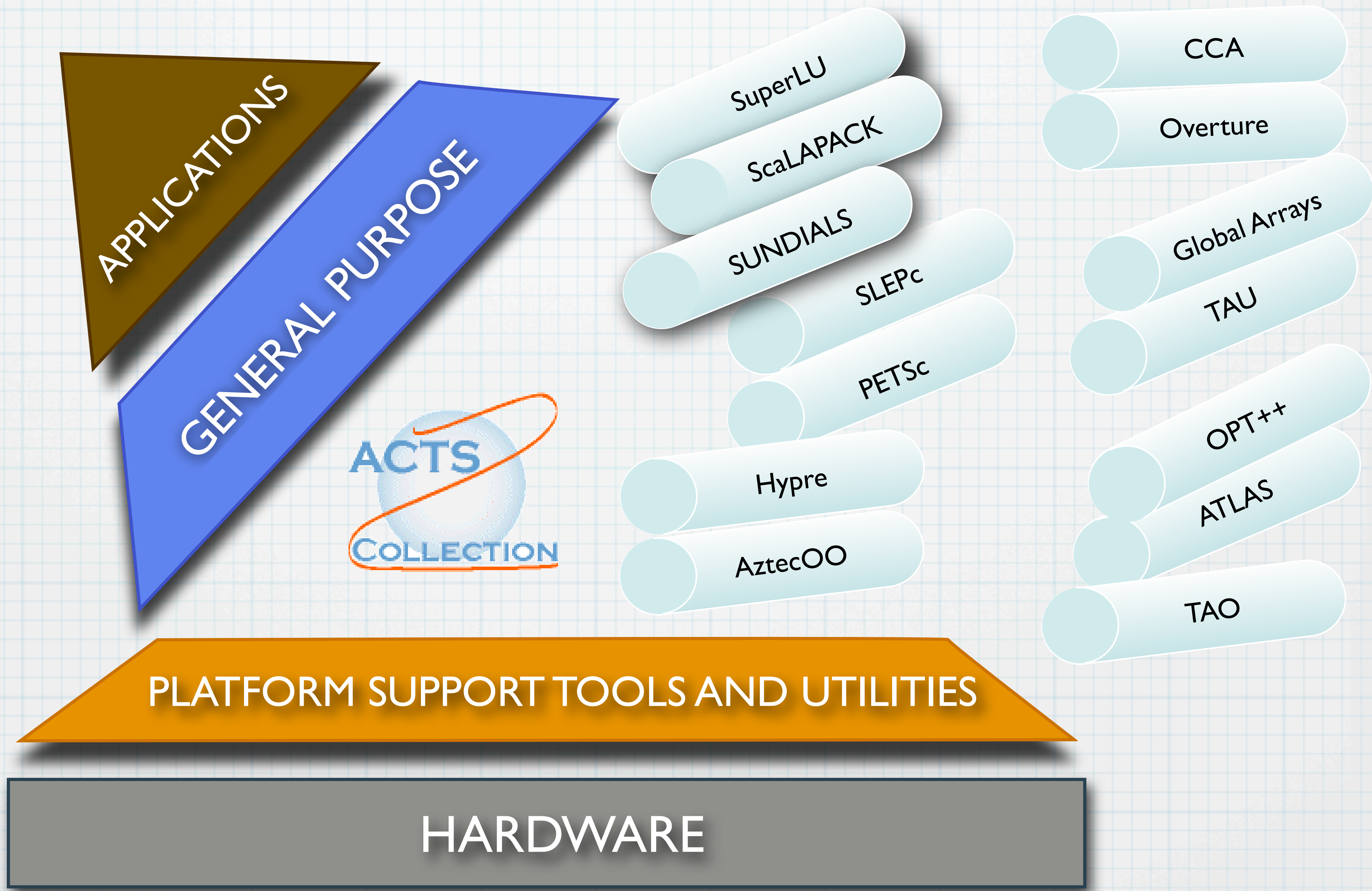


GENERAL PURPOSE TOOLS

PLATFORM SUPPORT TOOLS AND UTILITIES

HARDWARE

HPC Software Stack



Leveraging Sustainable Software

min[*time_to_first_solution*] (prototype)

min[*time_to_solution*] (production)

- Outlive Complexity
  - Increasingly sophisticated models
  - Model coupling
  - Interdisciplinary

(Software Evolution)

- Sustained Performance
  - Increasingly complex algorithms
  - Increasingly diverse architectures
  - Increasingly demanding applications

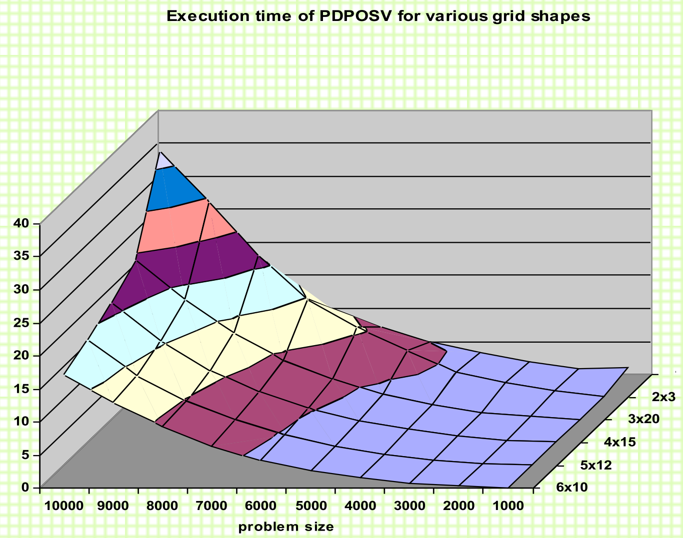
(Long-term deliverables)

min[*software-development-*

max[*software_life*] and **max**[*resource_utilization*]

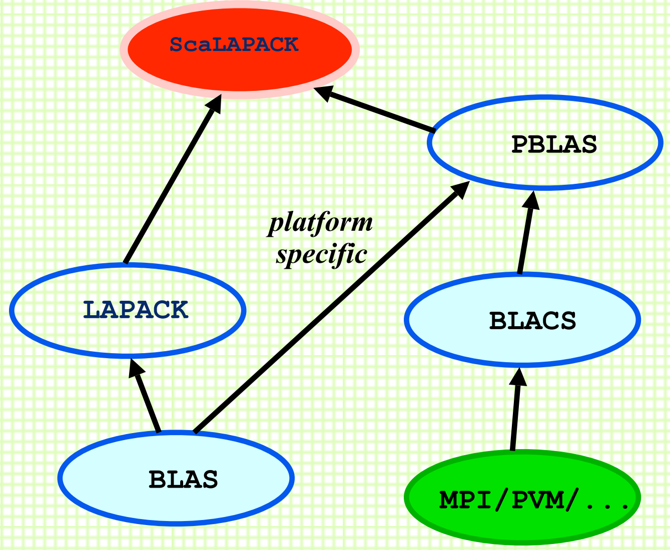
Our Approach to Software Sustainability

Profiling and Tracing Tools: TAU

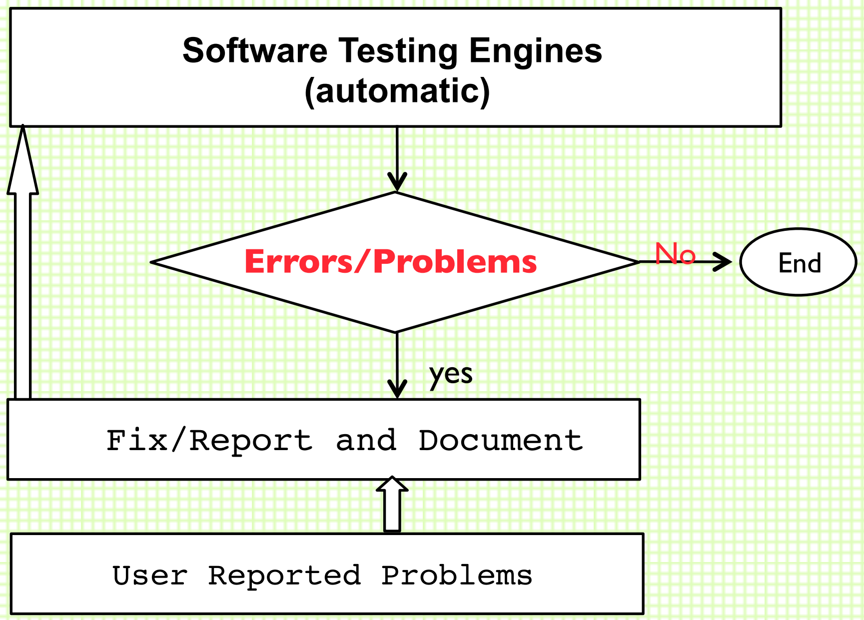


Performance and Scalability

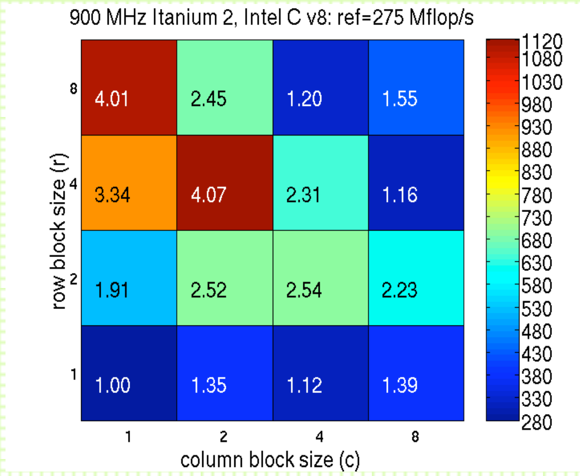
Software Dependency Graph



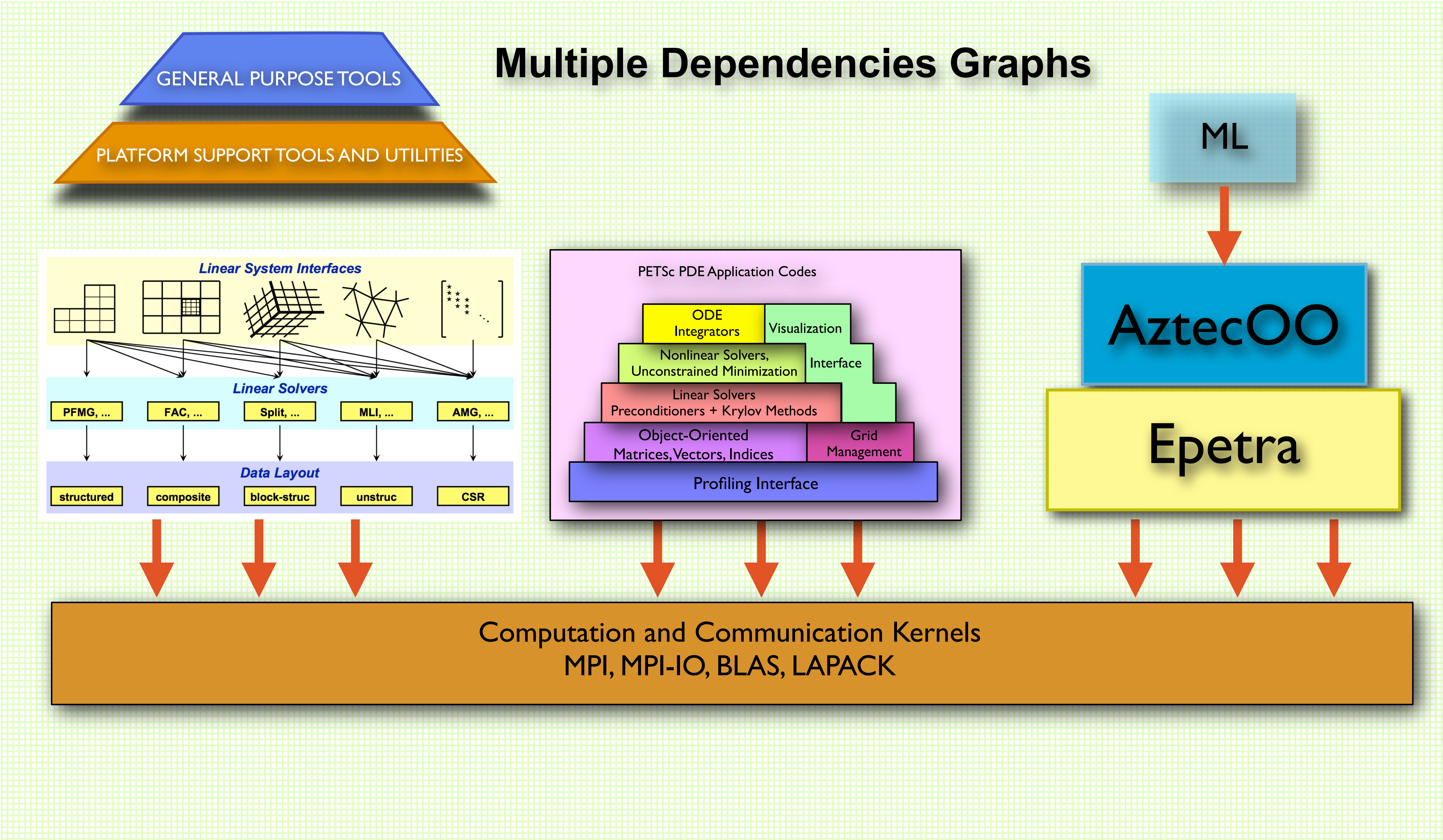
Automatic Testing and Verification



Auto-Tuning (OSKI, ATLAS,)



Software Dependency Graph



Outline

- ✦ Motivation
- ✦ Introduction to the DOE ACTS Collection
- ◆ Available functionality in the ACTS Collection

The U.S. DOE ACTS Collection

Category	Tool	Functionalities
Numerical	AztecOO	Scalable linear and non-linear solvers using iterative schemes.
	Hypre	A family of scalable preconditioners.
	PETSc	Scalable linear and non-linear solvers and additional support for PDE related work.
	OPT++	Object-oriented nonlinear optimization solvers.
	SUNDIALS	Solvers for the solution of systems of ordinary differential equations, nonlinear algebraic equations, and differential-algebraic equations.
	ScaLAPACK	High performance parallel dense linear algebra.
	SLEPc	Scalable algorithms for the solution of large sparse eigenvalue problems.
	SuperLU	Scalable direct solution of large, sparse, nonsymmetric linear systems of equations.
	TAO	Large-scale optimization software.
Code Development	Global Arrays	Supports the development of parallel programs.
	Overture	Supports the development of computational fluid dynamics codes in complex geometries.
	CCA	Forum and a framework for Software Interoperability
Run Time Support	TAU	Portable and scalable performance analyzes and tracing tools for C, C++, Fortran and Java programs.
Library Development	ATLAS	Automatic generation of optimized numerical dense algebra for scalar processors.

The U.S. DOE ACTS Collection

Category	Tool	Functionalities
Numerical	AztecOO	Scalable linear and non-linear solvers using iterative schemes.
	Hypre	A family of scalable preconditioners.
	PETSc	Scalable linear and non-linear solvers and additional support for PDE related work.
	OPT++	Object-oriented nonlinear optimization solvers.
	SUNDIALS	Solvers for the solution of systems of ordinary differential equations, nonlinear algebraic equations, and differential-algebraic equations.
	ScaLAPACK	High performance parallel dense linear algebra
	SLEPc	Scalable algorithms for the solution of large sparse eigenvalue problems.
	SuperLU	Scalable direct solution of large, sparse, nonsymmetric linear systems of equations.
	TAO	Large-scale optimization software.
Code Development	Global Arrays	Supports the development of parallel programs.
	Overture	Supports the development of computational fluid dynamics codes in complex geometries.
	CCA	Forum and a framework for Software Interoperability
Run Time Support	TAU	Portable and scalable performance analyzes and tracing tools for C, C++, Fortran and Java programs.
Library Development	ATLAS	Automatic generation of optimized numerical dense algebra for scalar processors.

http://acts.nersc.gov

ACTS COLLECTION

BERKELEY LAB

Office of Science
U.S. DEPARTMENT OF ENERGY

ACTS Numerical Tools: *Functionality*

Computational Problem	Methodology	Algorithms	Library
Systems of Linear Equations	Direct Methods	LU Factorization	ScaLAPACK(dense) SuperLU (sparse)
		Cholesky Factorization	ScaLAPACK
		LDL^T (Tridiagonal matrices)	ScaLAPACK
		QR Factorization	ScaLAPACK
		QR with column pivoting	ScaLAPACK
		LQ factorization	ScaLAPACK

ScaLAPACK

SuperLU

ACTS Numerical Tools: *Functionality*

Computational Problem	Methodology	Algorithms	Library
Systems of Linear Equations *(cont..)*	Iterative Methods	Conjugate Gradient	AztecOO (Trilinos) PETSc
		GMRES	AztecOO PETSc Hypre
		CG Squared	AztecOO PETSc
		Bi-CG Stab	AztecOO PETSc
		Quasi-Minimal Residual (QMR)	AztecOO
		Transpose Free QMR	AztecOO PETSc

ACTS Numerical Tools: *Functionality*

Computational Problem	**Methodology**	**Algorithms**	**Library**
Systems of Linear Equations *(cont..)*	Iterative Methods *(cont..)*	SYMMLQ	PETSc
		Precondition CG	AztecOO PETSc Hypre
		Richardson	PETSc
		Block Jacobi Preconditioner	AztecOO PETSc Hypre
		Point Jocobi Preconditioner	AztecOO
		Least Squares Polynomials	PETSc

PETSc

Hypre

Trilinos

ACTS Numerical Tools: *Functionality*

Computational Problem	**Methodology**	**Algorithms**	**Library**
Systems of Linear Equations (*cont..*)	Iterative Methods (*cont..*)	SOR Preconditioning	PETSc
		Overlapping Additive Schwartz	PETSc
		Approximate Inverse	Hypre
		Sparse LU preconditioner	AztecOO PETSc Hypre
		Incomplete LU (ILU) preconditioner	AztecOO
		Least Squares Polynomials	PETSc
	MultiGrid (MG) Methods	MG Preconditioner	PETSc Hypre
		Algebraic MG	Hypre
		Semi-coarsening	Hypre

ACTS Numerical Tools: *Functionality*

SLEPc

ScaLAPACK

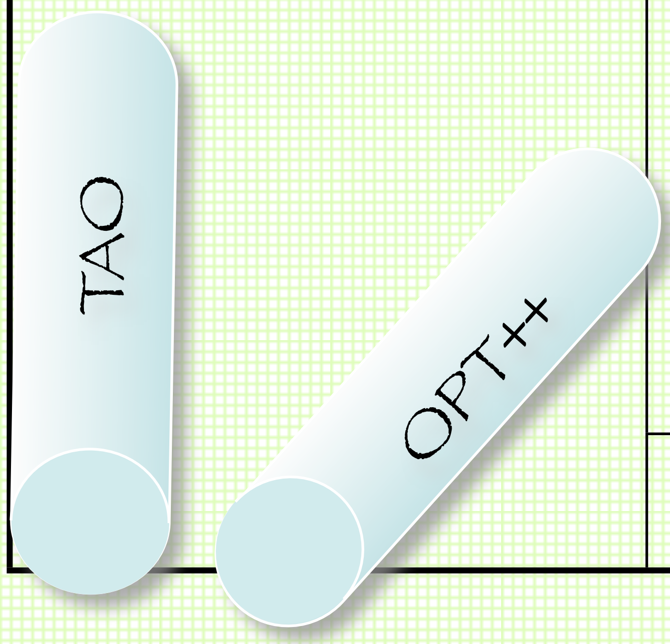
Computational Problem	Methodology	Algorithm	Library
Linear Least Squares Problems	Least Squares	$min_x \mid\mid b - Ax \mid\mid_2$	ScaLAPACK
	Minimum Norm Solution	$min_x \mid\mid x \mid\mid_2$	ScaLAPACK
	Minimum Norm Least Squares	$min_x \mid\mid b - Ax \mid\mid_2$ $min_x \mid\mid x \mid\mid_2$	ScaLAPACK
Standard Eigenvalue Problem	Symmetric Eigenvalue Problem	$Az = \lambda z$ *For* $A=A^H$ or $A=A^T$	ScaLAPACK (dense) SLEPc (sparse)
Singular Value Problem	Singular Value Decomposition	$A = U\Sigma V^T$ $A = U\Sigma V^H$	ScaLAPACK (dense) SLEPc (sparse)
Generalized Symmetric Definite Eigenproblem	Eigenproblem	$Az = \lambda Bz$ $ABz = \lambda z$ $BAz = \lambda z$	ScaLAPACK (dense) SLEPc (sparse)

ACTS Numerical Tools: *Functionality*

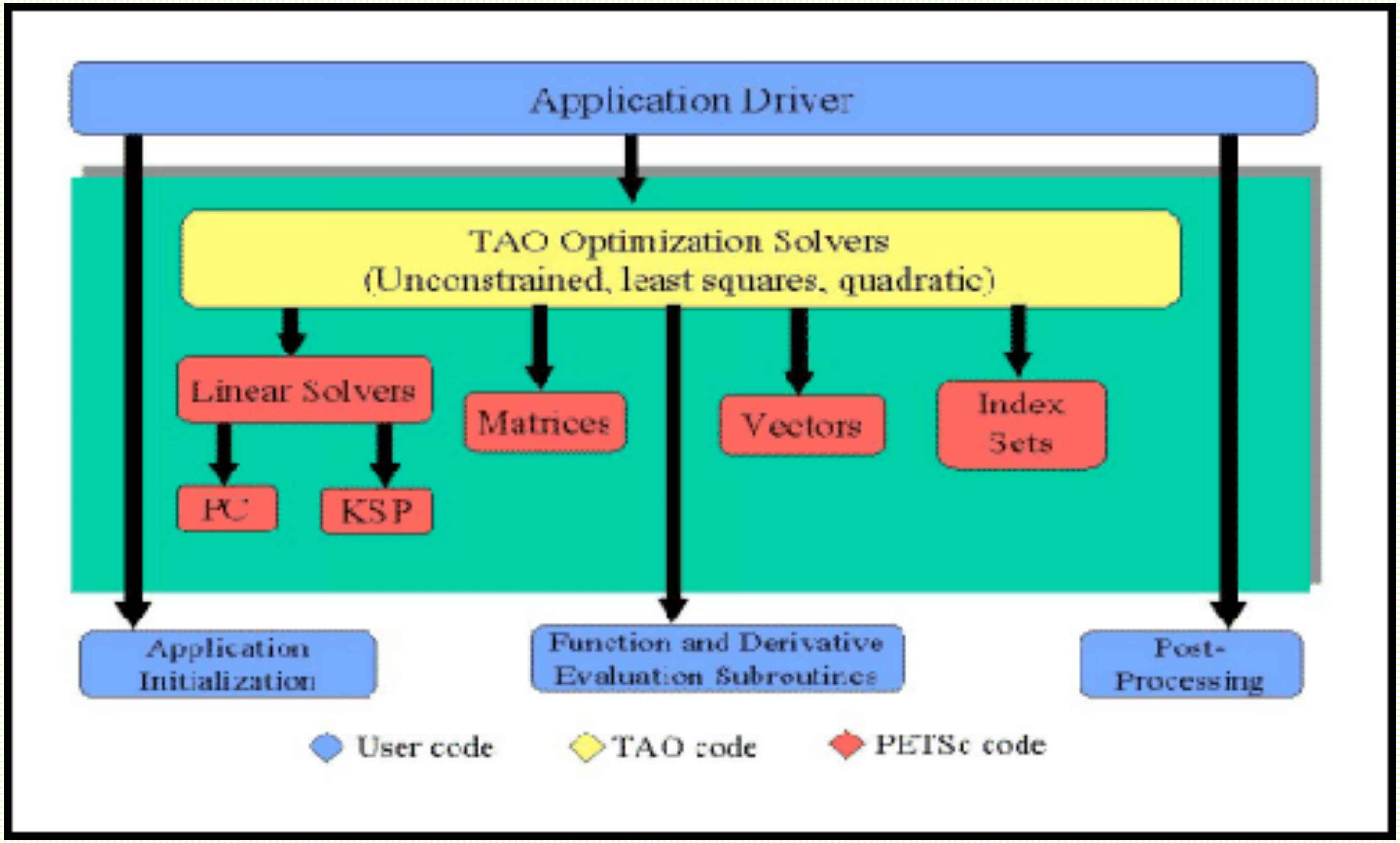
Computational Problem	**Methodology**	**Algorithm**	**Library**
Non-Linear Equations	Newton Based	Line Search	PETSc
		Trust Regions	PETSc
		Pseudo-Transient Continuation	PETSc
		Matrix Free	PETSc

ACTS Numerical Tools: *Functionality*

Computational Problem	Methodology	Algorithm	Library
Non-Linear Optimization	Newton Based	Newton	OPT++ TAO
		Finite-Difference Newton	OPT++ TAO
		Quasi-Newton	OPT++ TAO
		Non-linear Interior Point	OPT++ TAO
	CG	Standard Non-linear CG	OPT++ TAO
		Limited Memory BFGS	OPT++
		Gradient Projections	TAO
	Direct Search	No derivate information	OPT++



TAO - Interface with PETSc

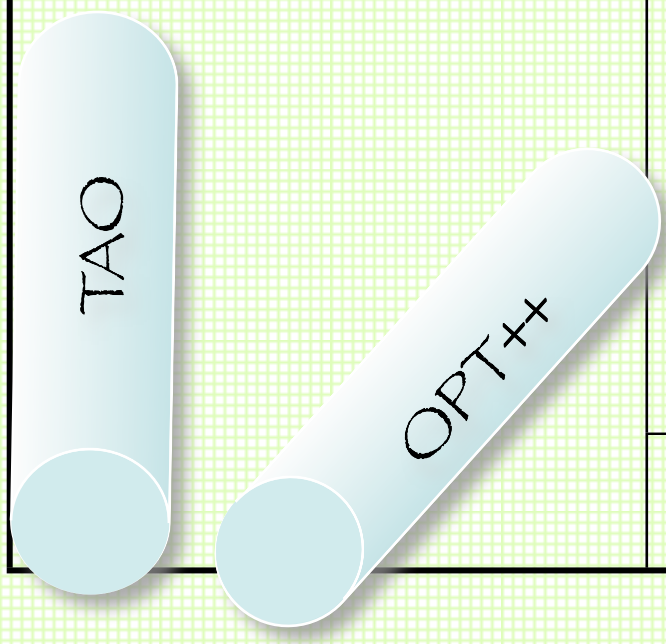


OPT++ Interfaces

- Four major classes of problems available
  - *NLF0(ndim, fcn, init_fcn, constraint)*
    - Basic nonlinear function, no derivative information available
  - *NLF1(ndim, fcn, init_fcn, constraint)*
    - Nonlinear function, first derivative information available
  - *FDNLF1(ndim, fcn, init_fcn, constraint)*
    - Nonlinear function, first derivative information approximated
  - *NLF2(ndim, fcn, init_fcn, constraint)*
    - Nonlinear function, first and second derivative information available

ACTS Numerical Tools: *Functionality*

Computational Problem	Methodology	Algorithm	Library
Non-Linear Optimization	Newton Based	Newton	OPT++ TAO
		Finite-Difference Newton	OPT++ TAO
		Quasi-Newton	OPT++ TAO
		Non-linear Interior Point	OPT++ TAO
	CG	Standard Non-linear CG	OPT++ TAO
		Limited Memory BFGS	OPT++
		Gradient Projections	TAO
	Direct Search	No derivate information	OPT++

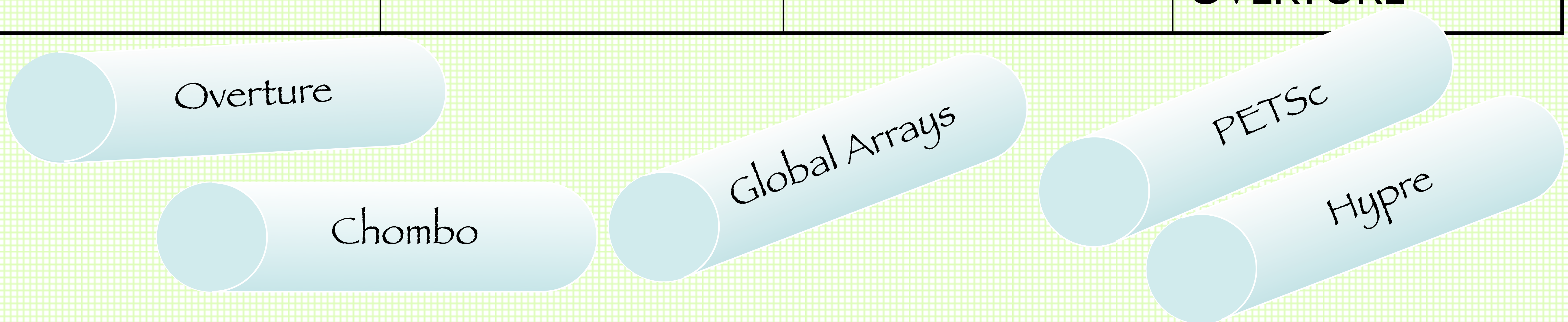


ACTS Numerical Tools: *Functionality*

Computational Problem	Methodology	Algorithm	Library
Non-Linear Optimization (cont..)	Semismoothing	Feasible Semismooth	TAO
		Unfeasible semismooth	TAO
Ordinary Differential Equations	Integration	Adam-Moulton (Variable coefficient forms)	CVODE (SUNDIALS) CVODES
	Backward Differential Formula	Direct and Iterative Solvers	CVODE CVODES
Nonlinear Algebraic Equations	Inexact Newton	Line Search	KINSOL (SUNDIALS)
Differential Algebraic Equations	Backward Differential Formula	Direct and Iterative Solvers	IDA (SUNDIALS)

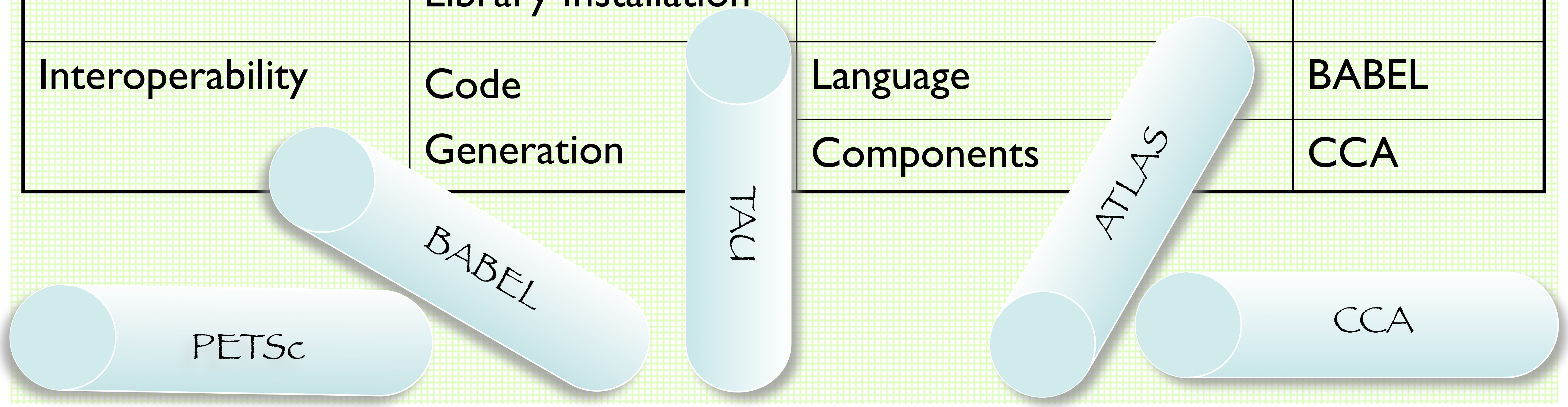
ACTS Tools: *Functionality*

Computational Problem	Support	Techniques	Library
Writing Parallel Programs	Distributed Arrays	Shared-Memory	Global Arrays
		Grid Generation	OVERTURE
		Structured Meshes	CHOMBO (AMR) Hypre OVERTURE PETSc
		Semi-Structured Meshes	CHOMBO (AMR) Hypre OVERTURE



ACTS Tools: *Functionality*

Computational Problem	Support	Technique	Library
Profiling	Algorithmic Performance	Automatic instrumentation	PETSc
		User Instrumentation	PETSc
	Execution Performance	Automatic Instrumentation	TAU
		User Instrumentation	TAU
Code Optimization	Library Installation	Linear Algebra Tuning	ATLAS
Interoperability	Code Generation	Language	BABEL
		Components	CCA



Outline

- Motivation
- Introduction to the DOE ACTS Collection
- Available functionality in the ACTS Collection
- Software sustainability

ACTS Collection → ACTS Software Sustainability Center

Long-Term Maintenance

Tools for Run-time Support:
- Scalable Debugging Tools
- Performance Monitoring Tools

Software Development Tools
Numerical Software
Library Optimization
Software Distribution
Software Dependency Graph:
- Platform, Basic and Interoperability

Independent Testing and Evaluation

Testing Platforms
Verification Engines
Computer Vendor Collaborations
Computational Sciences and Engineering Networking:
- Developers Exchange
- User Feedback
- Problem/Bug Tracking
- International Collaborations

DOE Leadership and Flagship Computing Facilities

ACTS TOOLS

Enabling Petascale

Outreach and Dissemination

ACTS Information Center:
- Guide to Available Services
- Technical Reports
- Newsgroups
- On-line Tutorials
- Uniform Tool Documentation
- Well Documented Examples for All Tools

Workshops
Short Courses and Coding Camps

High-Level User Support

High-Level User Interfaces
To Tool Users:
- Help with Tool Selection
- Help with Tool Utilization
- Help with Tool Installation
- Develop High-Level User Interfaces

To Tool Developers:
- Tool Long-term Maintenance Practices
- Tool Distribution Utilities, Licensing
- Tool Integration Mechanisms

Minimum Requirements for Sustainable Software

- **Robustness**
  - Maintained across platforms
  - Compiler independent
  - Precision Independent
  - Error Handling
  - Check Pointing

Minimum Requirements for Sustainable Software

- Robust
- **Scalable (across large Petascale systems)**

Minimum Requirements for Sustainable Software

- Robust
- Scalable
- **Extensible (New Algorithms, New Techniques)**

Minimum Requirements for Sustainable Software

- Robust
- Scalable
- Extensible
- **Interoperable**
  - Frameworks/PSE
  - Tool-to-Tool
  - **Component Technology**
    - **More Flexible**
    - **Retains better Robustness, Scalability, and Extensibility**
    - **Long term pay-offs**

http://www.cca-forum.org

Minimum Requirements for Sustainable Software

- Robust
- Scalable
- Extensible
- Interoperable
- **User Friendly Interfaces**
- **Well documented**

Minimum Requirements for Sustainable Software

```
CALL BLACS_GET( -1, 0, ICTXT )
 CALL BLACS_GRIDINIT( ICTXT, 'Row-major', NPROW, NPCOL )
:
 CALL BLACS_GRIDINFO( ICTXT, NPROW, NPCOL, MYROW, MYCOL )
:
:
 CALL PDGESV( N, NRHS, A, IA, JA, DESCA, IPIV, B, IB, JB, DESCB,
 $              INFO )
```

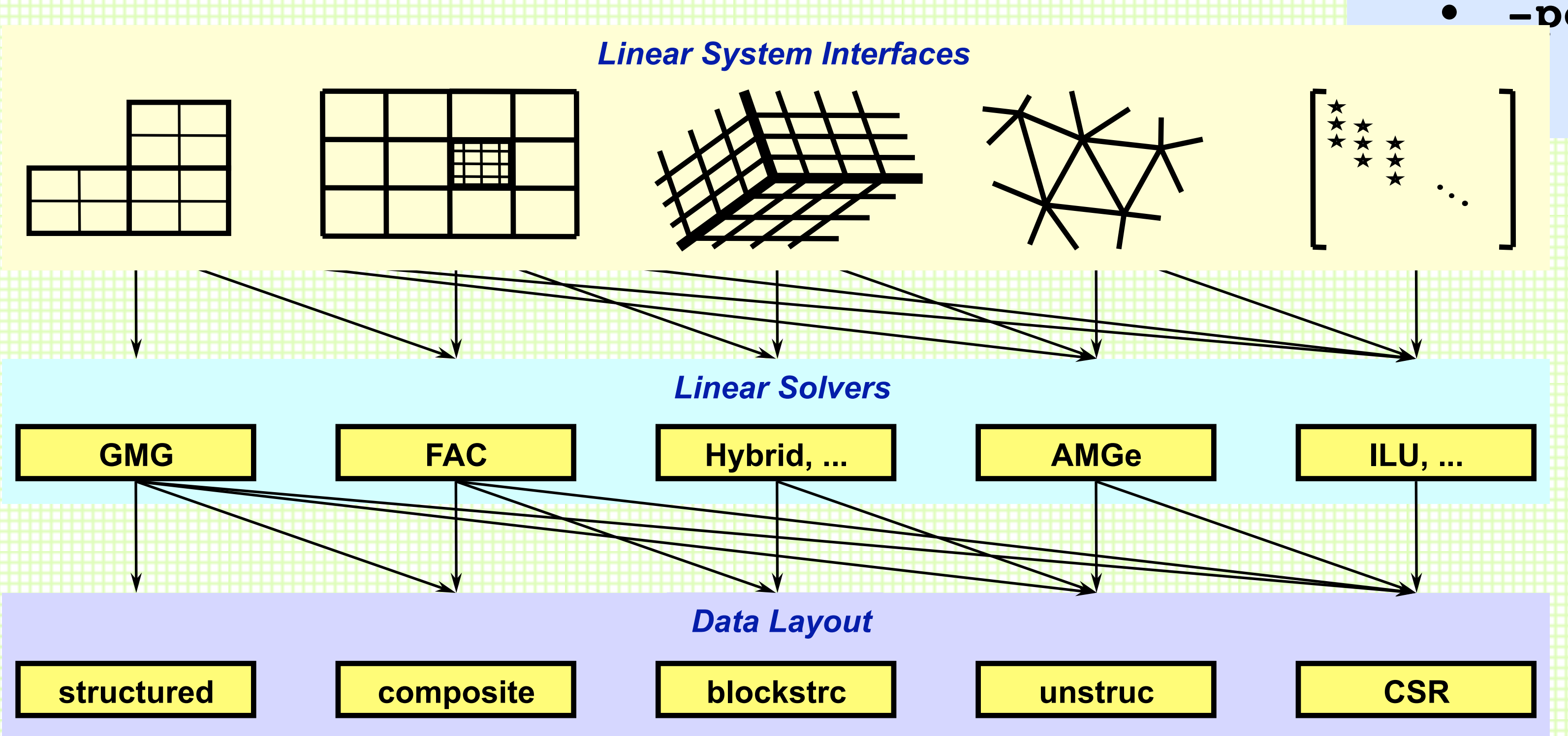
Library Calls

- -ksp_type [cg,gmres,bcgs,tfqmr,…]
- -pc_type [lu,ilu,jacobi,sor,asm,…]

More advanced:

- -ksp_max_it <max_iters>
- -ksp_gmres_restart <restart>
- -pc_asm_overlap <overlap>

Command lines



Problem Domain

Minimum Requirements for Sustainable Software

PyACTS

matlab*P

Star-P

NetSolve

User

View_field(T1)

$$Ax = b$$

$$Az = \lambda z$$

$$A = U\Sigma V^T$$

High Level Interfaces

OPT++ PAWS Globus CUMULVS TAU

AZTEC Hypre PETSc Chombo Global Arrays

ScaLAPACK SuperLU TAO PVODE Overture

Minimum Requirements for Sustainable Software

- Robust
- Scalable
- Extensible
- Interoperable
- User Friendly Interfaces
- Well documented
- **Periodic Tests and Evaluations**

Versions (tools, systems, O/S, compilers)

- Sanity-check (robustness)
- Interoperability (maintained)
- Consistent Documentation

Minimum Requirements for Sustainable Software

- Robust
- Scalable
- Extensible
- Interoperable
- User Friendly Interfaces
- Well documented
- Periodic Tests and Evaluations
- **Portability and Fast Adaptability (The Evolution)**

Outline

- Motivation
- Introduction to the DOE ACTS Collection
- Available functionality in the ACTS Collection
- Software sustainability
- This week at the ACTS Collection Workshop

This week at the Workshop

Tuesday	Wednesday	Thursday	Friday
Registration Opens 7:45 AM	*Doors Open* 8:00 AM	*Doors Open* 8:00 AM	*Doors Open* 8:00 AM
Welcome Remarks and Introduction **T. Drummond** 08:30 - 09:30	*PETSc* **S. Balay** 08:30 -10:30	*SuperLU* **S. Li** 08:30 - 09:30	*VisIt* **H. Childs** 08:30 - 10:30
ScaLAPACK **T. Drummond** 09:30 - 10:30		*Invited Talk* **H. Simon** CRD Div. Director 09:30 – 10:30	
Break 10:30 - 11:00	***Break*** 10:30 - 11:00	***Break*** 10:30 - 11:00	***Break*** 10:30 - 11:00
Hypre **R. Falgout** 11:00 – 12:00	*SLEPc* **J. Roman** 11:00 – 12:00	*TAU* **S. Shende** 11:00 – 12:30	*CCA* **CCA Team*** 11:00 – 12:30
Group Photo 12:00 - 12:30	*CITRIS Tour* **M. Nikravesh** 12:00 - 12:30		
Lunch 12:30 - 13:30	***Lunch*** 12:30 - 13:30	***Lunch*** 12:30 - 13:30	***Lunch*** 12:30 - 13:30

This week at the Workshop

Tuesday	Wednesday	Thursday	Friday
Zoltan **C. Chevalier** 13:30 – 14:30	*TAO* **J. Sarich** 13:30 – 14:30	*Trilinos* **J. Hu** 13:30 – 14:30	*CCA* *Hands-On* **CCA TEAM*** 13:30 – 16:30
Global Arrays **B. Palmer** 14:30 – 15:30	*Parallel I/O* **K. Antipas** 14:30 – 15:30	*Overture* **B. Henshaw** 14:30 – 15:30	
Break 15:30 - 16:00	***Break*** 15:30 - 16:00	***Break*** 15:30 - 16:00	
ScaLAPACK *Hands-On* **T. Drummond** 16:00 – 17:00	*PETSc* *Hands-On* **S. Balay** 16:00 – 17:00	*TAU* *Hands-On* **S. Shende** 16:00– 17:00	**Workshop Ends**
Global Arrays *Hands-On* **M. Krishnan** **B. Palmer** 17:00 – 18:00	*SLEPc* *Hands-On* **J. Roman** 17:00 – 18:00	*Trilinos* *Hands-On* **J. Hu** 17:00– 18:00	
	TAO *Hands-On* **S. Sarich** 18:00 – 19:00	*Overture* *Hands-On* **B. Henshaw** 18:00– 19:00	
Welcome Dinner **M. Nikravesh** 19:00 - 21:00			

*** CCA TEAM:**
David Bernholdt
Tamara Dahlgren
Wael Elwasif
Tom Epperly
Sameer Shende

This week at the Workshop

Hands-On:

- Your login name should be written in your badge
- Passwords:

  On Wheeler Hall PCs: c@1national

  On NERSC computers: DOE-09acts

This week at the Workshop

- **Picture today at lunch break outside this building!**
- **Return your Vouchers for travel!!**
- **Return your sign computer policy forms**

THANK YOU

Outline

- ✦ Motivation
- ✦ Introduction to the DOE ACTS Collection
- ✦ Available functionality in the ACTS Collection
- ✦ Software sustainability
- ✦ This week at the ACTS Collection Workshop
- Acknowledgements

Acknowledgments

- National Energy Research Scientific Computing Center (NERSC) for the use of their IBM SP (bassi) and Linux Cluster (jacquard)
- Yeen Mankin for all the great management and support running all the logistics of the workshop
- Cathy Sage for travel support
- To all the speakers and teams represented at the Tenth DOE ACTS Collection Workshop
- CITRIS for hosting the Tenth DOE ACTS Collection Workshop
- DOE Office of Science